# A Methodology for Determining the Dynamic Exchange of Resources in Nuclear Fuel Cycle Simulation

Matthew J. Gidden[a,b,*], Paul P. H. Wilson[b]

[a] *International Institute for Applied Systems Analysis, Schlossplatz 1, A-2361 Laxenburg, Austria*
[b] *University of Wisconsin - Madison, Department of Nuclear Engineering and Engineering Physics, Madison, WI 53706*

**Abstract**

Simulation of the nuclear fuel cycle can be performed using a wide range of techniques and methodologies. Past efforts have focused on specific fuel cycles or reactor technologies. The Cyclus fuel cycle simulator seeks to separate the design of the simulation from the fuel cycle or technologies of interest. In order to support this separation, a robust supply-demand communication and solution framework is required. Accordingly an agent-based supply-chain framework, the Dynamic Resource Exchange (DRE), has been designed implemented in Cyclus. It supports the communication of complex resources, namely isotopic compositions of nuclear fuel, between fuel cycle facilities and their managers (e.g., institutions and regions). Instances of supply and demand are defined as an optimization problem and solved for each timestep. Importantly, the DRE allows each agent in the simulation to independently indicate preference for specific trading options in order to meet both physics requirements and satisfy constraints imposed by potential socio-political models. To display the variety of possible simulations that the DRE enables, example scenarios are formulated and described. Important features include key fuel-cycle facility outages, introduction of external recycled fuel sources (similar to the current Mixed Oxide (MOX) Fuel Fabrication Facility in the United States), and nontrivial interactions between

*Corresponding author
*Email address:* gidden@iiasa.ac.at (Matthew J. Gidden)

fuel cycles existing in different regions.

*Keywords:* nuclear fuel cycle, optimization, agent-based modeling

---

## 1. Introduction

The nuclear fuel cycle (NFC) is a complex, physics-dependent supply chain of uranium and thorium ore based fuels, recycled materials (such as reprocessed uranium, plutonium, and other minor actinides), and final disposal of some subset of isotopes of transmuted material. Uranium is mined, milled, and enriched to some level based on the type and fuel management scheme (e.g., a 12 or 18-month refueling schedule) of the reactor which is being fueled. Used fuel can then be stored for a period of time before either being disposed of via interment or being utilized in a advanced fuel cycle by recycling its fissile and fertile isotopes. The ability to model such a system while maintaining physical consistency due to transmutation and isotopic decay is a challenging simulation problem. Through simulation, nuclear systems can be analyzed in order to support decision-making processes addressing a variety of goals, e.g., reducing system cost, future planning of storage facilities, studying the dynamics governing system transitions, and estimating long-term system sustainability.

NFC simulation is performed by a variety of actors, including governments, national laboratories, universities, international governance organizations, and consulting agencies. Accordingly, many modeling strategies have been applied, spanning a wide range of modeling detail for both nuclear facilities and fuel in order to obtain sufficient simulation functionality [1]. For instance, some simulators describe reactors by fleet (or type) and solve material balances for the entire fleet in aggregate [2, 3, 4, 5] while others instantiate individual (or discrete) facilities [6]. Similarly, some simulators make detailed calculations of fuel depletion due to reactor fluence [7, 8] whereas others use pre-tabulated values that depend (generally) on burnup values for thermal reactors and conversion ratios for fast reactors [4].

There are, broadly, three categories of concern to the design of an NFC simulator. The first is facility deployment, i.e., how, why, and when certain facilities are instantiated in the simulation. The most common reactor deployment mechanism allows a user to define an energy growth curve and, for each type of

reactor in the simulation, a percentage of that total energy demand to be met by that reactor type. It is also common for simulators to adjust deployments based on look-ahead heuristics of future material availability [9, 10]. The second design category is the fidelity with which the physical and chemical processes involved in the nuclear fuel cycle are modeled. Broadly, physical fidelity includes two processes, isotopic decay and isotopic transmutation due to fuel's residency in a reactor. To date, there is still disagreement as to the physical fidelity required to accurately capture sufficient system detail [11]. The third category concerns the communication of supply and demand between facilities, in other words, how facilities are connected in the simulation. In general, connections between facilities can either be static or dynamic and can either be fleet-based or facility-based. A static connection implies that material will always flow between two types of facilities, whereas a dynamic connection implies that a facility's input or output connection may change. Simulator design is dependent on the underlying modeling approach. For example, using system dynamics [12] naturally leads to a static, fleet-based approach [2, 3, 4], whereas developing a stand-alone, discrete event or time simulation [13] can lead to higher levels of modeling fidelity in areas of concern [6, 8, 7].

Cyclus, a NFC simulator developed by the CNERG team at the University of Wisconsin, was designed to support different levels of model fidelity at different portions of the fuel cycle [14]. By Law's definition [13], Cyclus is a dynamic, discrete-event simulation that uses a fixed-increment time advance mechanism. Its design seeks to separate the design concerns of the three categories described above, supporting, for example, both fleet and individual facility models and allowing for either exogenous or endogenous facility deployment [15]. Further, one of the primary goals of Cyclus is to separate the simulation environment from the specific fuel cycle or process being modeled. As such, the accuracy of any simulation will depend on the accuracy of the specific facility models being employed in that simulation.

However, a common infrastructure defining the method of facility connection and allowing communication between entities in the simulation is required. This

infrastructure must be flexible in order to support different approaches to each of the categories of simulation design. To do so, it must allow for static simulation entities (e.g., facilities) as well as dynamic entities that enter and exit the simulation. Further, it must support the changing of relationships between those entities based on simulation state. Finally, it must allow for communication of complex resource types, e.g., isotopic fuel vectors that change with time.

This work describes a novel approach to addressing this complicated series of design problems associated with the exchange of resources in a dynamic, physics-dependent, supply-chain simulation. It combines methods of both discrete-event simulation and agent-based modeling with an optimization approach to determine the constrained transfer of resources. Inspiration for the entity communication framework was taken from the existing agent-based supply-chain modeling literature [16, 17, 18, 19, 20] which provides a natural methodological fit to the present use case. Given time-dependent supply and demand of nuclear fuel, a version of the constrained, multi-commodity transportation problem is solved to determine resource transfers within a simulation time-step.

The remainder of this paper is structured as follows. Section 2 describes in detail the communication framework, optimization problem formulation, and possible solution techniques. Section 2.5 also describes a new archetype in the Cyclus ecosystem that utilizes this framework to enable entity relationships to drive material routing decisions. Section 3 then describes a series of scenarios that display the enhanced modeling capabilities enabled by this new simulation framework. Finally, section 4 provides concluding remarks and observations, reflecting on potential future work and use cases.

## 2. Methodology & Implementation

Dynamic Resource Exchange (DRE) is a inter-simulation, optimization-based methodology for determining transactions between suppliers and consumers. The core solution strategy is agnostic to resource types. The DRE is designed to support fuel cycle simulation, which is highly dependent on specific resource

properties (material isotopic vectors), through its agent communication framework. Because the communication framework can be specialized to any abstract resource type, the methodology and framework can be adapted to other complex supply chains.

The DRE enables the constrained transaction of complex resources between entities in a simulation given a measure of cardinal preference for each potential transactions. The full formulation of the description of supply and demand in the fuel-cycle context is denoted the Nuclear Fuel Cycle Transportation Problem (NFCTP), a variant of the classic family of transportation problems in optimization. Suppliers and consumers provide information about their supply and demand during an initial information gathering phase. Complex constraints can be supplied during this phase. Supply and demand is then translated into a resource-agnostic *exchange graph*. The graph can be solved feasibly with a heuristic or optimally by translating it into a mixed integer-linear program. Given a solution, final trades are constructed and executed. In order to provide a more concrete discussion, all descriptions of the DRE and its mechanisms assume an exchange based on nuclear materials, the particular type of resource most important to a fuel-cycle simulation context.

Section 2.1 begins by providing a short overview of the classic optimization tools on which this work is built. An outline of the DRE methodology's progression with respect to the simulation architecture is described in section 2.2. Section 2.3 then details the interface that agents within the simulation have with the DRE in order to communicate supply and demand information. A description of the DRE's graph-based and formulation based definitions and solution techniques is provided in section 2.4. Finally, section 2.5 describes a new `Region` archetype in the Cyclus ecosystem that utilizes the DRE to enable the *in situ* modeling of inter-state trade instruments, such as tariffs.

This section represents the culmination of significant previous effort [21, 22, 23]. What follows constitutes the refinement of previous descriptions of the DRE methodology with lessons learned from initial implementation and usage.

### 2.1. Multicommodity Transportation Problems

Supply and demand in a nuclear fuel cycle context is inherently a multicommodity problem: a light water reactor can be fueled by both uranium oxide (UOX) and mixed oxide (MOX) fuel, for instance. How it is fueled is a result of fuel availability, associated preferences, and its operating history and current fuel management strategy. In order to allow for complex physical and chemical constraints on both processes and inventories, an optimization-based approach is used which employs economics-based proxies to arrive at a solution of resource transfers within a simulation time-step.

The DRE translates agent supply and demand into a version of the Multicommodity Transportation Problem (MTP) [24] which belongs to the network flow family of optimization problems. A network flow problem is represented by a graph, $G(N, A)$, comprises nodes $N$ and arcs $A$. If flow can occur between some node $i$ and some other node $j$, then it flows along arc $(i, j)$. Given a graph instance, optimal flow between nodes can be found provided *objective coefficients* and *constraints*. *Decision variables* for this optimization problem comprise the optimal *flow assignment*. If all decision variables are linear, then the resulting formulation is termed a Linear Program (LP). If some decision variables are integer (e.g., binary), the formulation is termed a Mixed-Integer Linear Program (MILP).

Transportation problems model the flow of a commodity between source nodes and sink nodes which can have supply and demand constraints. A more complex transportation-problem formulation can support systems in which supply or demand can be met by multiple commodities. There is a unit cost $c_{i,j}^h$ for commodity $h$ to traverse arc $(i, j)$. A supplier of commodity $h$ has a certain supply capacity $s_i^h$ which cannot be surpassed and consumers of commodity $h$ have a certain demand level which must be met, $d_i^h$.

In the simplest extension from the single-commodity to multi-commodity transportation problem, arc constraints for all commodities are combined, i.e., there is a single capacity $u_{i,j}$ for a given arc $(i, j)$. A classic application of this

enhanced complexity deals with data networks. Multiple classifications of data exist, but they all must traverse the same network infrastructure. Accordingly, the infrastructure can only accommodate a certain quantity of total flow among all communication types. The formulation of the multi-commodity flow problem is shown in Equation 1. Note the commodity coupling in Equation 1d.

$$\min_{x} \sum_{i\in I}\sum_{j\in J}\sum_{h\in H} c^h_{i,j}x^h_{i,j} \tag{1a}$$

$$\text{s.t.} \sum_{i\in I} x^h_{i,j} \geq d^h_j \qquad \forall\, j \in J, \forall\, h \in H \tag{1b}$$

$$\sum_{j\in J} x^h_{i,j} \leq s^h_i \qquad \forall\, i \in I, \forall\, h \in H \tag{1c}$$

$$\sum_{h\in H} x^h_{i,j} \leq u_{i,j} \qquad \forall\, (i,j) \in A \tag{1d}$$

$$x^k_{i,j} \geq 0 \qquad \forall\, (i,j) \in A, \forall\, h \in H \tag{1e}$$

The proceeding sections describe how agent state at a given point in time in a simulation is translated into constraint and cost parameters (e.g., $s$, $d$, and $c$ in Equation 1). While the DRE-based formulation is not a direct mirror of Equation 1, it is a useful basis of comparison. Arcs in the DRE-based formulation are identified by their commodity and grouped into appropriate constraints, whereas the multicommodity nature arises in Equation 1d as a constraint on individual arcs.

### *2.2. Communication between Simulation and Formulation*

Defining a robust interface for the communication between simulation and optimization components is one of the most difficult aspects in constructing a single, combined framework. A clear mapping of simulation and agent concepts to formulation coefficients, constraints, and parameters is required. Conceptually, the DRE implements this interface in three layers as shown in Figure 1.

The first layer includes information for specific `Resource` [1] types. For example,

[1] Terms that directly map to names of C++ classes in the Cyclus code base are formatted

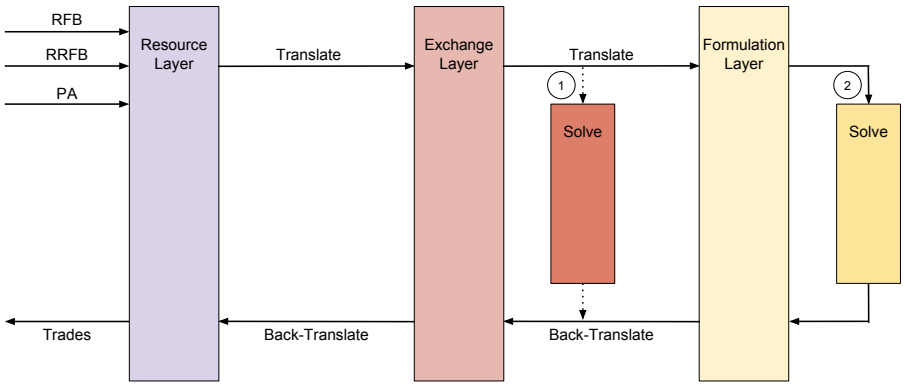



Figure 1: The full DRE workflow is shown. The information gathering phase, described in section 2.3, results in the resource layer. The resource layer is translated to the exchange layer; marked by the number 1, a decision is made whether to continue translation or to directly solve the instance, as described in section 2.4.4. If the exchange is not solved, it is translated into an instance of the NFCTP resulting in the formulation layer as shown in section 2.4.5. A choice of solver is made, marked by the number 2, and the instance is solved. The solution is back-translated through the exchange and resource layers. The result is a series of resource trades to be executed in the simulation.

a `Material`-based exchange is used for agents to communicate supply and demand information regarding `Material` objects. The *resource layer* is the point of entry and exit of the DRE framework. It is the agent-facing interface of the DRE: supply and demand is provided to the DRE as input during the information gathering step, and trades to be executed are provided to agents as output.

The second layer, called the *exchange layer*, is a `Resource`-agnostic representation of supply and demand. Supply/demand constructs in the first layer are translated into stateful objects representing nodes, arcs, constructs that carry constraint information, *et cetera*. The collection of objects and structures combine to create an `ExchangeGraph`. Any custom, CYCLUS-aware solver can be applied to an `ExchangeGraph` to determine a feasible solution to the DRE.

In order to use sophisticated, 3rd party LP and MILP solving libraries, the `ExchangeGraph` must be translated into an appropriate data structure representing an instance of the NFCTP, resulting in the *formulation layer*. The Open Solver Interface (OSI) [25] is used to create the necessary formulation structures, including a constraint matrix and objective coefficient vector. The NFCTP instance is then solved.

After a feasible, perhaps optimal, solution to the NFCTP is found, whether in the exchange or formulation layer, the solution is back-translated to the resource layer. The agents associated with successful supply-demand connections are informed, and trades of resources between agents are executed.

### *2.3. Agent Interaction with the DRE*

#### *2.3.1. Supply and Demand*

The DRE begins with three *phases*, the terminology of which is influenced from previous supply chain agent-based modeling work [17]. Importantly, this information-gathering step is agnostic as to the supply-demand matching algorithm used, it is concerned only with querying the current status of supply and demand in the simulation. The collective information gathering procedure is

as shown for clarity.

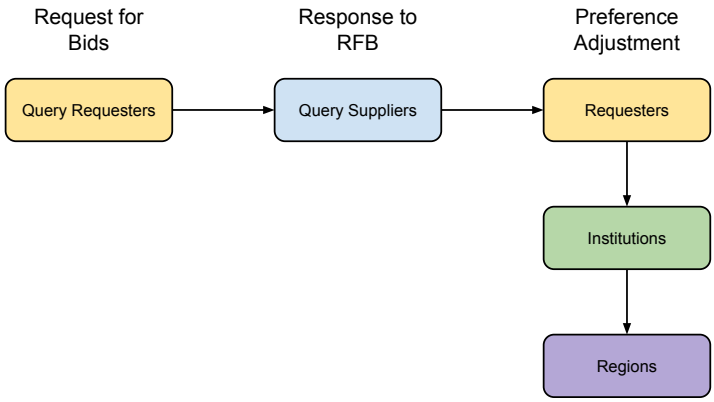



Figure 2: Schematic illustrating the DRE's information gathering phases: Request for Bids (RFB), Response to Request for Bids (RRFB), and Preference Adjustment (PA).

shown in Figure 2.

The first phase allows consumers of commodities to denote both the quantity of a commodity they need to consume as well as the target isotopics, or quality, by *posting* their demand to the market exchange. This posting informs producers of commodities what is needed by consumers, and is termed the *Request for Bids* (RFB) phase. Consumers are allowed to over-post, i.e., request more quantity than they can actually consume, as long as a corresponding capacity constraint accompanies this posting. Requests can be denoted as *exclusive*. An exclusive request is one that must either be met in full or not at all. Exclusive requests allow the modeling of quantized, packaged transfers, e.g., fuel assemblies.

Consumers are allowed to post demand for multiple commodities that may serve to meet the same capacity, and the collection of all commodities requested is termed its *request portfolio*. For example, consider a light water reactor (LWR) that can be filled with MOX or UOX. It can post a demand for both, but must define a preference over the set of possible commodities that can be consumed. Such requests are termed *mutual requests*. Another example is that of an advanced fuel fabrication facility, i.e., one that fabricates fuel partially from separated material that has already passed through a reactor. Such a facility can choose to fill the remaining space in a certain assembly with various types of fertile

material, including depleted uranium from enrichment or reprocessed uranium from separations. Accordingly, it could demand both commodities as long as it provides a corresponding constraint with respect to total consumption. A set of exclusive requests may also be grouped as mutual requests, in which case the set is termed *mutually exclusive*.

At the completion of the RFB phase, the market exchange will have a set of request portfolios. Each portfolio consists of a set requests. Arbitrary constraints over the set of requests can be provided that are functions of quantity, $x$, or quality, $q$. When communicating constraining information, agents must provide a total constraining quantity, $b$, and a *constraint coefficient conversion function*, $\beta(q_{i,j})$. Utilizing this information, constraints similar to Equation 1b can be constructed for all possible trades $A_j$ for requester $j$,

$$\sum_{(i,j)\in A_j} \beta(q_{i,j}) x_{i,j} \geq b. \tag{2}$$

Each request additionally has an associated preference. For requests that mutually satisfy a given demand, a preference distribution over those requests informs the solver as to which should be satisfied first, given the constraints. Finally, each request portfolio has a specific quantity associated with it.

The second phase allows suppliers to *respond* to the set of request portfolios, and is termed the *Response to Request for Bids* (RRFB) phase (analogous to Julka's Reply to Request for Quote phase [17]). Each request portfolio comprises requests for some set of commodities. For each request, suppliers of that commodity denote production capacities and an isotopic profile of the commodity they can provide. Suppliers are allowed to offer the null set of isotopics as their profile, effectively providing no information of the offer's chemistry or physics. Suppliers are also allowed to denote responses as exclusive, as is done in the RFB phase. This functionality again supports the notion of quantized orders, e.g., in the case of fuel assemblies. Supply responses can also be grouped into mutual responses, and sets of responses may be mutually exclusive. The full collection of responses for a given supplier is denoted as its *supply portfolio*.

A supplier may have its production constrained by communicating the same information as consumers. Constraints corresponding to Equation 1c are constructed in the same manner as Equation 2. Suppliers can provide one or more constraints. For example, a processing facility may have both a throughput constraint (i.e., it can only process material at a certain rate) and an inventory constraint (i.e., it can only hold some total material). Further, the facility could have a constraint on the quality of material to be processed, e.g., it may be able to handle a maximum radiotoxicity for any given time step which is a function of both the quantity of material in processes and the isotopic content of that material. Multiple of such constraints are allowed. At the completion of the RRFB phase the possible connections between supplier and producer facilities, i.e., the arcs in the graph of the transportation problem, have been established with specific capacity constraints defined both by the quantity and quality of commodities that will traverse the arcs.

#### *2.3.2. Preferences*

The final phase of the information gathering procedure allows consumer facilities to adjust their set of preferences and for managers of consumer facilities to affect the consumer's set of preferences. Accordingly, the last phase is termed the *Preference Adjustment* (PA) phase. Socio-economic models are allowed to inform the exchange of resources in this phase by allowing a facility's higher-level decision makers, e.g., the region in which a facility resides, to also adjust preferences. For example, a region can detect a trans-regional trade between one of its facilities and a facility in another region. If a tariff model is employed, the trade preference and be diminished or even removed.

For facilities, preference adjustment provides a mechanism to act with arbitrary complexity in response to offers provided by producer facilities. Consider the example of a reactor facility that requests two fuel types, MOX and UOX, and receives two responses to its request for MOX, each with different isotopic profiles. It can then assign preference values over the set of potential MOX providers. Repositories provide another prime example where preference adjust-

ment can be naturally employed. A repository may have a defined preference of material to accept based upon its heat load or radiotoxicity, both of which are functions of the quality, or isotopics, of a material. In certain simulators, limits on fuel entering a repository are imposed based upon the amount of time that has elapsed since the fuel has exited a reactor, which can be assessed during this phase. The time constraint is, in actuality, a constraint on heat load or radiotoxicity (one must let enough of the fission products decay). A repository could analyze possible input fuel isotopics and set the arc preference of any that violate a given rule to 0, thereby eliminating that arc.

The game theoretic notion of preferences can be quite useful in fuel cycle simulation. Specifically, cardinal utility, or cardinal preferences [26] provides a relative measure of preference such that any two preferences can be directly compared, provided an arbitrary scaling, similar to the comparison of costs in a system. The notion of preference also nicely extends the work of Oliver's affinity metric [27]. Additionally, costs in a nuclear fuel cycle simulation have reasonably large uncertainty [28] and are generally applied to the output of a simulator as a post-processing step. Therefore, preferences, a proxy of cost, can be used to drive consistent decision-making within a simulation.

There exists a body of literature that examine Nash Equilibria in the context of optimal flow models [29, 30, 31]. However, the complexity of such models quickly brings them out of the scope of the needs for dynamic modeling of multi-lateral scenarios ranging 100+ years in a "reasonable" amount of computation time.

#### *2.3.3. Trades*

The information gathering phase defines the population of *potential* trades. The system is then solved by either a heuristic or optimization solver, resulting in a feasible set of *finalized* trades. The DRE is completed at a given time step when trades are *executed* by instructing all supplier agents to send their finalized trades to consumer agents.

### 2.4. The Nuclear Fuel Cycle Transportation Problem

An instance of supply and demand defined by the DRE's information gathering step is cast to a constrained, bipartite network which represents a variant of the MTP, entitled the *Nuclear Fuel Cycle Transportation Problem* (NFCTP). It can be solved by any heuristic that provides a feasible solution to such networks are valid. For this work, a greedy heuristic is designed and implemented. The system can be solved optimally, however, by formulating the system as a mathematical program, and a MILP formulation is provided. Formulated as a MILP, the system can be solved with any available solver. `COIN-CBC` [32], a popular open-source branch-and-bound solver, is used in this work.

#### 2.4.1. Exchange Graph

Objects and data structures generated in the information gathering procedure are used in the formal definition of the NFCTP by mapping the agent-supplied information onto a *bipartite* graph. This mapping allows for the translation from the resource to exchange layers shown in Figure 1. Information is mapped to properties of arcs, which represent proposed trades, and portfolio-based node groupings. The components of an exchange graph have a one-to-one mapping with simulation entities. For example, nodes in the graph represent distinct bids and requests provided by agents.

Each supply and request portfolio can be considered separately, i.e., there is no information shared between two portfolios of a given agent. The set of supply portfolios is denoted as $S$ and the set of request portfolios is denoted as $R$; each agent may have multiple portfolios in a given exchange. Each supply portfolio comprises $s_M$ supply nodes, and each request portfolio comprises $r_N$ nodes. For notation simplicity, nodes within portfolios are referred to with single indices (e.g., $i$ or $j$), and collections of arcs (connections between supply and request nodes) associated with a given portfolio are referred to as $(i, j) \in A_s$ and $(i, j) \in A_r$, for supply and request portfolios respectively.

For each request node, $j$, there may be many bid nodes; however, there is a one-to-one mapping between bid nodes and request nodes. In other words, a

given bid node, $i$, is a unique response to a request node, $j$. Because of defined constraints, there may not be sufficient supply in the simulated exchange. To ensure a feasible solution, an unconstrained false supply node is added to the exchange graph. Additionally, false nodes are added to each request portfolio and are connected to the false supply source. These arcs are denoted as *false arcs*. Figure 3 shows a fully defined exchange graph. As an example, $A_s$ in this example is defined as $\{(i,j),(i',j')\}$.

In the bipartite graph, portfolios act as partitions that group nodes together. Each portfolio has a set of commodities, $H$, associated with it. These are denoted $H_s$ for supply portfolios and $H_r$ for request portfolios. Node groups share a set of common constraints, $K$, and request node groups share a common notion of satisfiable quantity, i.e., a default mass-based constraint. Each constraint has a constraining value, $b_s^k$ and $b_r^k$, respectively.

Additionally, each portfolio and constraint has a defined constraint coefficient conversion function, denoted $\beta_s^k$ for supply portfolios and $\beta_r^k$ for request portfolios. Request portfolios are provided a mass constraint by default for which coefficients are unity and whose constraining value is is $b_r^x$. If requested commodities are labeled as *mutual*, then a weighting coefficient is generated for each request in the mutual set, $M$, in order to support cases where different commodities are requested with different quantities. The coefficient is defined by the ratio between the the average request quantity over all mutual requests and $x_m$

$$\beta_{r,m} = \frac{\overline{x_M}}{x_m}, \tag{3}$$

The constraint conversion functions are utilized in the NFCTP by applying them to the proposed resource transfers, creating constraint coefficients. Coefficients for supply constraints are defined as

$$a_{i,j}^k = \beta_s^k(q_{i,j}). \tag{4}$$

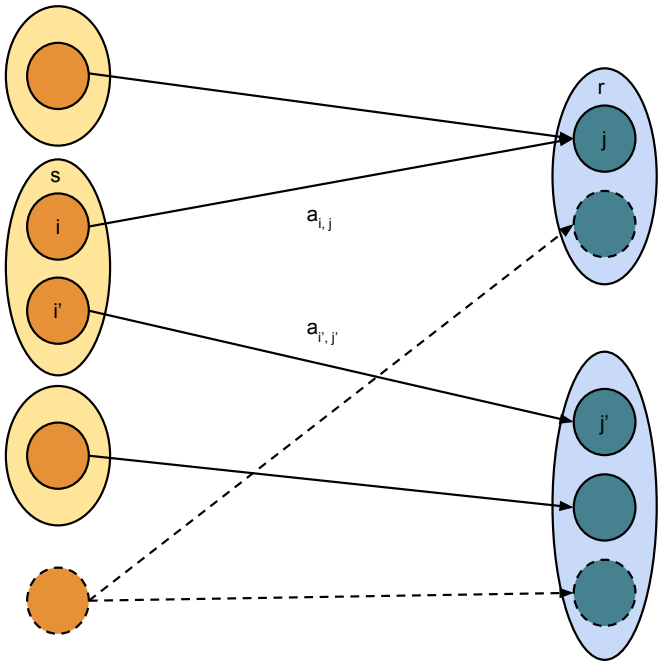



Figure 3: An example exchange with supply nodes colored orange on left and request nodes colored blue on right. As shown, there can be multiple supply nodes connected to a request node, but each supply node corresponds uniquely to one request node. It is a specific response to that request, as outlined in the RRFB phase. In this example, there are three supplier agents and two consumer agents. The second consumer has two requests (for different commodities) which may satisfy its demand. The second supplier can supply the commodities requested by both consumers and has provided two bids accordingly. The false supplier and consumer nodes are shown with a dashed outline. Similarly, false arcs are dashed. Note that the false nodes have no associated portfolio structure – there are no constraints associated with false nodes and arcs. The inclusion of a false supplier and consumer guarantees a feasible solution.

Coefficients for request constraints are defined as

$$a_{j,i}^{k} = \beta_{r}^{k}(q_{i,j}). \tag{5}$$

Finally, for each supply-request node pair, there is an associated preference, $p_{i,j}$. The set of all preferences is denoted $P$. Similarly, flow between a node pair is denoted $x_{i,j}$, and the set of all flows is denoted $X$. The possible flow on an arc is provided an upper bound by the request node quantity, $\widetilde{x_j}$.

In any network flow problem, the objective coefficients associated with transporting commodities drive the solution. Given the nature of supply and demand constraints, the transportation problem naturally lends itself to a minimum cost formulation. A preference-based formulation has been presented thus far due to the difficulties of employing reasonable cost coefficients. While directly using costs should be available to users, in practice using a more abstract notion of preferences is simpler.

Formally, a preference function, $p_{i,j}(h)$, is defined which is a cardinal preference ordering over a consumer's satisfying commodity set.

$$p_{i,j}(h) \ \forall i \in I \ \forall h \in H_r \tag{6}$$

A preference is assigned to each arc in the NFCTP and is a function both of the consumer, $j$, and producer, $i$, and the quality, $q_{i,j}$, of the proposed resource transfer from consumer to producer. The dependence on producer encapsulates the relationship effects due to managerial preferences. The preference set used in the NFCTP formulation follows directly from the Preference Adjustment phase described in section 2.3. A cost translation function, $f$, is defined that operates on the commodity preference function to produce an appropriate cost for the NFCTP.

$$f : p_{i,j}(h) \rightarrow c_{i,j} \tag{7}$$

For the purposes of this work, any operator that preserves preference monotonicity and cardinal ordering is suitable. The inversion operator has been chosen because

it preserves required features and also allows for easy translation from preference to cost as well as translation from cost to preference.

$$f(x) = \frac{1}{x} \tag{8}$$

The preferences given to each false arc, $p_f$, is defined to be lower than the lowest preference in the system, $P$.

$$p_f < \min P \tag{9}$$

Because preferences are defined as in Equation 9, any false arc will only be engaged if no other possible arc can be engage, due to capacity constraints. If any flow is assigned to false arcs after the exchange graph is solved, that flow is ignored when initiating transactions.

If cost data and a valid cost assignment methodology is developed in the future, costs may be used directly, and the preference-to-cost translation may be ignored.

*2.4.2. An Example Exchange Graph*

During the information gathering step in section 2.3, consumers and suppliers are queried based on *commodities*. A consumer is allowed to request multiple commodities, and a supplier is allowed to supply multiple commodities. However, each possible resource transfer, i.e., each arc, is based on a single commodity. Accordingly, it is possible to color each arc, given a commodity-to-color mapping.

For example, consider the exchange graph shown in Figure 4 with two fuel commodities ($A$, $B$), two requesters ($R_1$, $R_2$), and three suppliers ($S_1$, $S_2$, $S_3$) in the configuration described by Tables 1 and 2. The resulting exchange graph can be colored as shown in Figure 4.

The notion of commodities is critical during the information gathering step as it is the basic classification used in communicating supply and demand. It is also useful when an exchange graph is formed, because the graph may be able to be partitioned by collections of commodities. However, once minimally

| Supplier | Commodities |
|---|---|
| $S_1$ | $A$ |
| $S_2$ | $A$, $B$ |
| $S_3$ | $B$ |

Table 1: A mapping from suppliers to commodities supplied.

| Consumer | Commodities |
|---|---|
| $R_1$ | $A$ |
| $R_2$ | $B$ |

Table 2: A mapping from requesters to commodities requested.

connected exchange graphs are established, solution mechanisms do not employ the notion of commodities. Rather, quantities, constraints, and preferences are used.

*2.4.3. Communicating Constraints*

Constraint coefficients are determined for an arc based on the proposed resource to be transferred along that arc, the requester's constraint conversion functions, and the suppliers constraint conversion function. Consider a supplier enrichment facility, $s$, which produces the commodity enriched uranium (EU). This facility has two constraints on its operation for any given time period: the amount of Separative Work Units (SWU) that it can process, $b_s^{\mathrm{SWU}}$, and the total natural uranium (NU) feed it has on hand., $b_s^{\mathrm{NU}}$. Note that neither of these capacities are measure directly in the units of the commodity it produces, i.e., kilograms of EU. The constraint set for $s$ is then

$$K_s = \{\mathrm{SWU}, \mathrm{NU}\}. \tag{10}$$

Consider a set of requests for enriched uranium that this facility can possibly meet. Such requests have, in general, two parameters: $P_j$, the total product

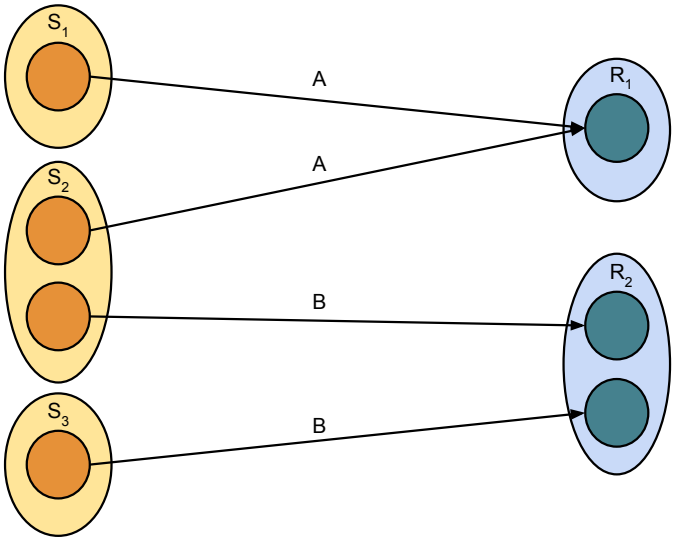



Figure 4: The same exchange shown in Figure 3 with arcs and portfolios labeled based on Tables 1 and 2.

quantity (in kilograms), and $\varepsilon_j$, the product enrichment (in w/o $^{235}$U).[2] For the purposes of this constraint set, the quality of material in question is its enrichment, i.e.,

$$q_j \equiv \varepsilon_j. \tag{11}$$

These values are set during the information-gathering phase of the overall matching algorithm, and can therefore be considered constant. Further, note that, in general, an enrichment facility's operation, or rather its capacity, is governed by two parameters: $\varepsilon_f$, the fraction of $^{235}$U in its feed material, and $\varepsilon_t$, the fraction of $^{235}$U in its tails material. These parameters determine the amount of SWU required to produce some amount of enriched uranium, shown in Equation 12 as well as the amount of natural uranium, or feed, required, as

[2]The notation for enrichment, $\varepsilon_j$, is chosen over its normal form, $x_p$, to limit confusion with the notation of material flow, $x_{i,j}$.

shown in Equation 13.

$$\begin{aligned} SWU = \; & P(V(\varepsilon_j) + \frac{\varepsilon_j - \varepsilon_f}{\varepsilon_f - \varepsilon_t} V(\varepsilon_t) \\ & - \frac{\varepsilon_j - \varepsilon_t}{\varepsilon_f - \varepsilon_t} V(\varepsilon_f)) \end{aligned} \tag{12}$$

$$F = P \frac{\varepsilon_j - \varepsilon_t}{\varepsilon_f - \varepsilon_t} \tag{13}$$

$P$ in Equations 12 and 13 is the amount of produced enriched uranium, $F$ is the amount of feed, or natural uranium, and $V(x)$ is the value function,

$$V(x) = (1 - 2x) \ln \left( \frac{1 - x}{x} \right) \tag{14}$$

435 Utilizing the above equations, one can denote the functional forms of the arguments of this facility's two capacity constraints.

$$\beta_s^{\mathrm{NU}}(\varepsilon_j) = \frac{\varepsilon_j - \varepsilon_t}{\varepsilon_f - \varepsilon_t} \tag{15}$$

$$\begin{aligned} \beta_s^{\mathrm{SWU}}(\varepsilon_j) = \; & V(\varepsilon_j) \\ & + \frac{\varepsilon_j - \varepsilon_f}{\varepsilon_f - \varepsilon_t} V(\varepsilon_t) \\ & - \frac{\varepsilon_j - \varepsilon_t}{\varepsilon_f - \varepsilon_t} V(\varepsilon_f) \end{aligned} \tag{16}$$

These constraints correspond to the per-unit requirements for enriched uranium of natural uranium feed and SWU. Finally, we can form the set of constraint equations for the enrichment facility by combining Equations 11, 15, and 16.

$$\sum_{j \in J} \beta_s^{\mathrm{NU}}(\varepsilon_j) \, x_{s,j} \leq b_s^{\mathrm{NU}} \tag{17}$$

$$\sum_{j \in J} \beta_s^{\mathrm{SWU}}(\varepsilon_j) \, x_{s,j} \leq b_s^{\mathrm{SWU}} \tag{18}$$

### *2.4.4. A Heuristic Solution*

Given an exchange graph, including false arcs, a feasible solution can be found. By definition a feasible solution is a *solution* to the possible flow of resources, but not necessarily an *optimal* solution. Many heuristics may be applied to bipartite graphs with constrained flows. A simple *greedy* heuristic is presented here and implemented.

The maximum flow along an arc, $\widehat{x_{i,j}}$, depends on the constraints associated with each node on the arc. For nodes $i$ and $j$ belonging to portfolios $s$ and $r$, respectively, the maximum allowable flow is defined as

$$\widehat{x_{i,j}} = \min\{\min\{\frac{b_s^k}{a_{i,j}^k}\ \forall k \in K_s\},\ \min\{\frac{b_r^k}{a_{i,j}^k}\ \forall k \in K_r\}\}. \tag{19}$$

The Greedy Exchange Heuristic, described in Algorithm 1 , matches maximum flow along arcs, up to the requested amount defined by each request portfolio, $b_r^x$, after having sorted all arcs. The constraining values of each arc, $b_k$, are updated upon declaration of a match (via an `AddMatch` function) .

The heuristic naturally accounts for mutual requests and exclusive trades, the two unique properties associated with the formulation. Mutual requests are accounted for by the weighted mass-balance constraint. Exclusivity is accounted for through initial screening (supply-request pairs that will not match are removed from the exchange in a pre-screening step) and the use of the maximum value function.

### *2.4.5. Mixed Integer Linear Programming Formulation*

The mathematical formulation NFCTP can be constructed by combining the components of an exchange graph and adding appropriate parameters and variables, translating from the exchange layer to the formulation layer as shown in Figure 1. The NFCTP is formulated as a mixed integer-linear program (MILP), rather than a linear program (LP) in order to allow for quantized commodity transfers that commonly arise in the fuel cycle context, such as the case of reactor fuel orders, which comprise a large amount of material orders within the

**Data**: A resource exchange graph with constraints and preferences.

**Result**: A valid set of resource flows.

sort request partitions by average preference;

**forall the** $r \in R$ **do**

  sort requests by average preference;

  matched $\leftarrow 0$;

  **while** *matched* $\leq b_r^x$ *and* $\exists$ *a request* **do**

    get next request;

    sort incoming arcs by preference;

    **while** *matched* $\leq b_r^x$ *and* $\exists$ *an arc* **do**

      get next arc;

      remaining $\leftarrow b_r^x$ - matched;

      to_match $\leftarrow \min\{\text{remaining}, \widehat{x_{i,j}}\}$;

      `AddMatch`(arc, to_match);

      matched $\leftarrow$ matched + to_match;

    **end**

  **end**

**end**

**Algorithm 1:** Greedy Exchange Heuristic

simulation context. By introducing binary decision variables, fuel orders can be guaranteed to be met by a single supplier, rather than allowing mixing of orders between potential suppliers. Similarly, it also guarantees that used fuel is sent to a single back-end facility, rather than being split between multiple facilities.

In order to simplify the formulation and maintain consistency with the exchange layer description, variables and parameters are referred to by their arc index, $(i, j)$. Sets of arcs are associated with suppliers $A_s$ are all arcs leaving a given supply portfolio, and sets of arcs are associated with requesters $A_r$ are all arcs entering a given request portfolio.

A binary decision variable $y_{i,j}$ is defined for each arc and has a value of 1 if flow occurs between producer node $i$ and consumer node $j$. If flow occurs,

its quantity will be equal to the equivalent flow upper bound along that arc, $\tilde{x}_j$. Binary variables, representing quantized flow, are directly related to the notion of *exclusive* bids and requests discussed in section 2.3. In the MILP formulation, an arc $(i, j)$ is considered exclusive if either node $i$ or node $j$ was defined as exclusive in the information gathering phase of the DRE. Given the set of arcs $A$, a partition exists such that $A$ can be separated into exclusive arcs and non-exclusive arcs, or arcs that allow partial flow, for each supplier and requester.

$$A = \bigcup_{r \in R} A_{p_r} \cup A_{e_r} \tag{20}$$

$$A = \bigcup_{s \in S} A_{p_s} \cup A_{e_s} \tag{21}$$

*Mutually exclusive* requests and responses, described in section 2.3, are defined as a set of requests or responses, of which only one may be satisfied. This is represented in the formulation as a constraint on the associated $y_{i,j}$ variables: only one arc in a mutually exclusive set may have a value of 1. The set of mutually exclusive arcs is denoted $M_s$ and $M_r$ for suppliers and requesters, respectively. The associated constraints are then defined by Equations 22 and 23.

$$\sum_{(i,j) \in M_s} y_{i,j} \leq 1 \; \forall \; s \in S \tag{22}$$

$$\sum_{(i,j) \in M_r} y_{i,j} \leq 1 \; \forall \; r \in R \tag{23}$$

Using the described arc partition notation allows for a much simpler written formulation of the MILP. The full formulation of the NFCTP is shown in Equation 24. The sets and variables involved in Equation 24 are described in Tables 3 and 4.

$$\min_{x,y} \; z \;=\; \sum_{(i,j)\in A_p} c_{i,j}x_{i,j} \;+\; \sum_{(i,j)\in A_e} c'_{i,j}y_{i,j} \tag{24a}$$

$$\text{s.t.} \sum_{(i,j)\in A_{p_r}} a^k_{i,j}x_{i,j} \;+\; \sum_{(i,j)\in A_{e_r}} a^{k\prime}_{i,j}y_{i,j} \geq b^k_r \qquad \forall\, k \in K_r, \forall\, r \in R \tag{24b}$$

$$\sum_{(i,j)\in M_r} y_{i,j} \leq 1 \qquad \forall\, r \in R \tag{24c}$$

$$\sum_{(i,j)\in A_{p_s}} a^k_{i,j}x_{i,j} \;+\; \sum_{(i,j)\in A_{e_s}} a^{k\prime}_{i,j}y_{i,j} \leq b^k_s \qquad \forall\, k \in K_s, \forall\, s \in S \tag{24d}$$

$$\sum_{(i,j)\in M_s} y_{i,j} \leq 1 \qquad \forall\, s \in S \tag{24e}$$

$$x_{i,j} \in [0, \tilde{x_j}] \qquad \forall\, (i,j) \in A_p \tag{24f}$$

$$y_{i,j} \in \{0, 1\} \qquad \forall\, (i,j) \in A_e \tag{24g}$$

A simplified representation of constraint coefficients for binary variables shown in Equation 25 and objective coefficients shown in Equation 26 is used.

$$a^{k\prime}_{i,j} = a^k_{i,j}\tilde{x_j} \tag{25}$$

$$c'_{i,j} = c_{i,j}\tilde{x_j} \tag{26}$$

### *2.5. Inter-region Policy Instruments*

Supporting economic and social models is rare among simulators. Only one simulator purports to have any endogenous economic decision making [10]. Modeling international fuel cycles requires a simulator to support a notion of regional boundaries. To date, DESAE is the only simulator to advertise such a feature, providing static models of regional relationships as input [33]. Accordingly, no NFC simulator provides any representation dynamic models of inter-region policy instruments, such as tariffs.

Cyclus natively supports inter-regional flows via its Region-Institution-Facility hierarchy [14]. While the DRE is capable of *supporting* dynamic relationship models through its preference adjustment phase, no such models have

| Set | Description |
|---|---|
| $S$ | suppliers (i.e., supply portfolios) |
| $R$ | requesters (i.e., request portfolios) |
| $A_{p_s}$ | arcs that allow *partial* flows for supplier $s$ |
| $A_{e_s}$ | *exclusive* flow arcs for supplier $s$ |
| $A_{p_p}$ | arcs that allow *partial* flows for requester $r$ |
| $A_{e_p}$ | *exclusive* flow arcs for requester $r$ |
| $M_s$ | arcs $(i, j)$ associated with *mutually exclusive* supply for supplier $s$ |
| $M_r$ | arcs $(i, j)$ associated with *mutually exclusive* requests for requester $r$ |
| $X$ | the feasible set of flows between producers and consumers |
| $Y$ | the binary variable set of flows between producers and consumers |

Table 3: Sets Appearing in the NFCTP Formulation

| Variable | Description |
|---|---|
| $c_{i,j}$ | the unit cost of flow from producer node $i$ to consumer node $j$ |
| $x_{i,j}$ | a decision variable, the flow from producer node $i$ to consumer node $j$ |
| $y_{i,j}$ | a decision variable, whether flow exists from producer node $i$ to consumer node $j$ |
| $a_{i,j}^k$ | the constraint coefficient for constraint $k$ on flow between nodes $i$ and $j$ |
| $b_s^k$ | the constraining value for constraint $k$ of supplier $s$ |
| $b_r^k$ | the constraining value for constraint $k$ of requester $r$ |
| $\tilde{x_j}$ | the requested quantity associated with request node $j$ |

Table 4: Variables Appearing in the NFCTP Formulation

heretofore been implemented. The only flow-based relationship models currently offered occur at the facility level. That is, certain facilities may set commodity-based preferences for potential material flows. For example, a `Reactor` prototype may set its preference for MOX-based fuels higher than UOX-based fuels, and

the DRE will provide it with MOX-based fuels if it is able.

A new `Region` archetype has been developed to explicitly support both static and dynamic inter-region policy instrument models. Named the `TariffRegion`, it applies instrument models, such as tariffs, during the preference-adjustment phase of the DRE according to a given *rule*. Rules may be applied, updated, and removed as a function of time, thereby supporting dynamic instrument models. Static models are trivially supported by applying a rule at the initial time step and not removing it.

Rules comprise *conditions* and *tariffs*. Given a condition and tariff, preferences are adjusted as shown in Algorithm 2. A rule's condition may depend on any factor that is query-able during the preference adjustment phase of the DRE. During PA, each potential resource transfer is known. Therefore, rules may depend on information regarding the supplier or consumer (e.g., in which region each resides), the commodity associated with the transfer, and both the resource quantity and quality (e.g., the fissile plutonium content for material resources).

**Data**: A potential trade, a condition, and a tariff value, $x$.

**Result**: An updated preference value, $p$.

**if** *trade meets condition* **then**
| return $p * x$

**else**
| return $p$

**end**

**Algorithm 2:** `TariffRegion` Preference Adjustment

## 3. Experimentation & Results

A number of computational experiments are conducted to highlight unique features enabled by the DRE in Cyclus. Each experiment is performed by solving instances of the DRE using both the `Greedy` heuristic and to optimality with the branch-and-bound solver `COIN-CBC`. A UOX-MOX one-pass recycle system with all required fuel cycle facilities is taken as the `base-case` scenario in order to

reduce the complexity of the fuel cycle and highlight departures from available simulators. For simplicity of demonstration, reactors are assumed to refuel completely with a single commodity rather than a combination of fuel types as is done in practice. A simulation time frame of 50 years is chosen with one-month time-steps (totaling 600 simulation time steps), sufficient to display all relevant effects. The nominal parameters of all common facilities in the simulation are shown in [34].

The `base-case` scenario is not process constrained (i.e., it is constrained only by the dynamics of Pu availability in the recycling stream). Reactors are allowed to be fueled by either UOX or MOX, with a preference for MOX over UOX, and refuel one-third of their total core mass every 18 months. Spent UOX fuel is allowed to be recycled, whereas spent MOX fuel is sent directly to a repository. In order to involve dynamism in the simulation, the population reactors grows linearly over time at a rate of 1 reactor every 5 years. An initial population of 20 reactors are deployed individually in each of the first 20 time-steps of the simulation as shown in Figure 5. Note that deployments are staggered in the initial period in order to avoid supply/demand clustering effect. A diagram of the full `base-case` fuel cycle is shown in Figure 6.

Three perturbations from the `base-case` scenario are used to provide examples of modeling capability enabled through the use of the DRE. The scenarios are summarized in Table 5 below and described in more detail in the following sections.

### *3.1. Separations Outage: Fuel Cycles with Supply Disruption*

The DRE provides a unifying framework in which any instance of supply and demand can be formulated and solved. This flexibility lends itself well to dynamic simulation in which the state of actors in a simulation, by definition, can change as the simulation progresses. In order to show case the types of simulations that are enabled by this feature, a fuel cycle simulation is constructed that has multiple types of reactor fuel input and a defined supply disruption within the recycled-fuel supply chain.

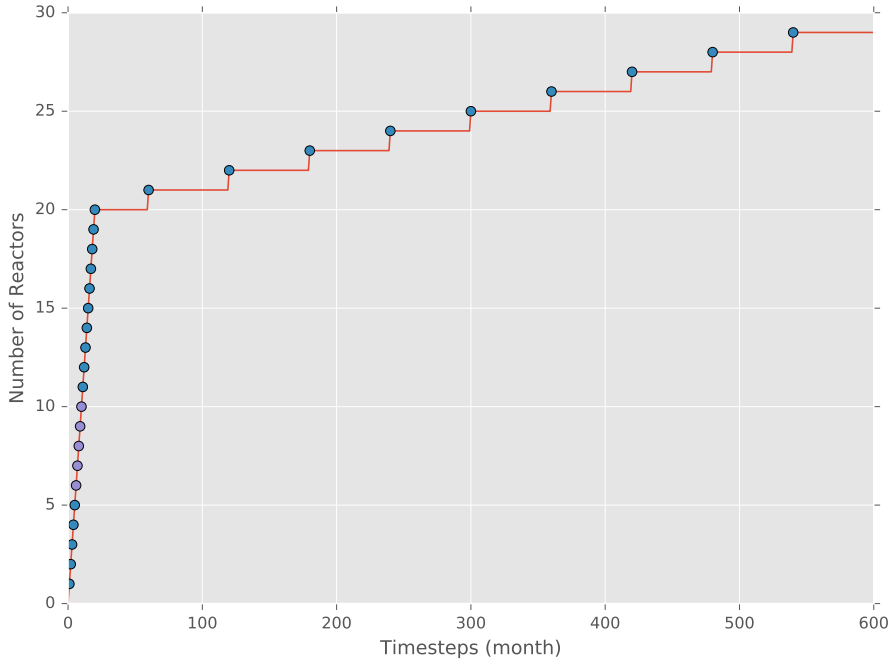



Figure 5: `Reactor` deployment in each simulation as a function of simulation time steps. Each point in the graph is a reactor being deployed in the simulation. Deployments for the `tariff` scenario are distinguished by color: blue represents deployments in Region A and purple represents deployments in Region B.

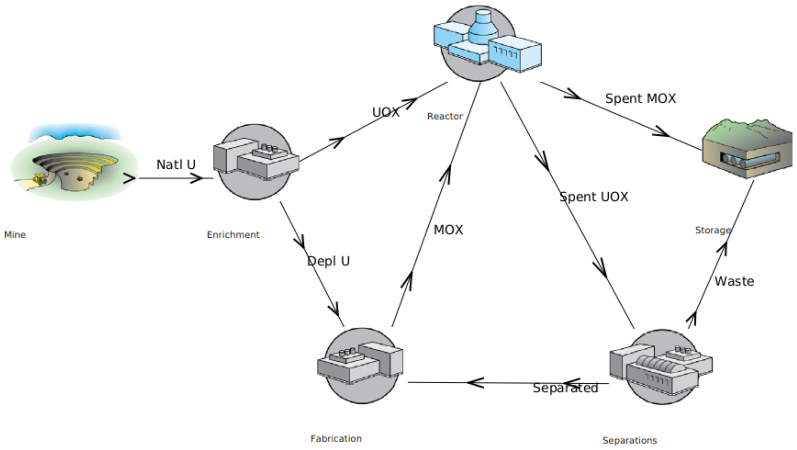



Figure 6: Material routing between in the `base-case` scenario, single-pass MOX fuel cycle. Possible arc flows are labeled with commodity names.

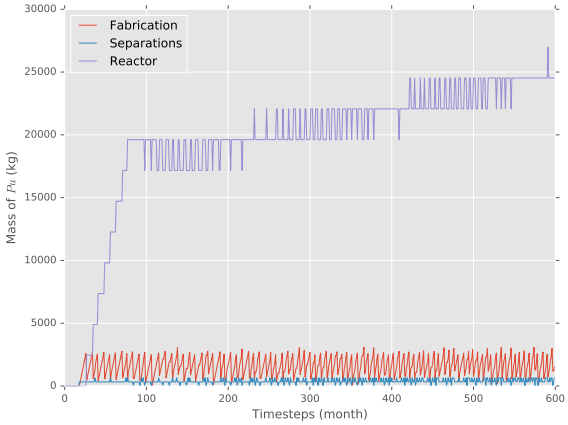
Table 5: Short Descriptions of Scenarios Ran

| **Scenario Name** | **Scenario Handle** | **Primary Departure from Base Case** | **Capability Highlighted** |
|---|---|---|---|
| Separations Outage | `outage` | Separations facility halts operation mid-simulation | System flexibility to recycling facilities operation |
| External MOX Supplier | `external` | An additional supplier of MOX enters mid-simulation | System flexibility to entry and exit of commodity suppliers |
| Regional Tariffs | `tariff` | Two regions are modeled with dynamic trade relationships | Ability to model nontrivial international relationships |

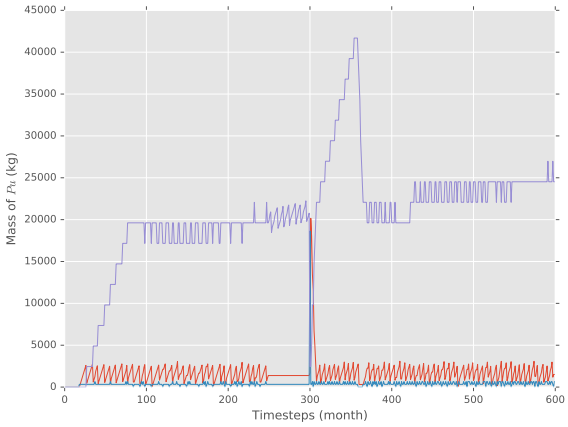
The chosen disruption is an outage of the separations facility shown in Figure 6. The outage begins at $t_i = 250$, lasts 50 time steps, ending at $t_f = 300$. During the outage, the remaining facilities in the supply chain operate normally, and the flow of fuel into and out of reactors adapts according to the state of available fresh fuel. Importantly, neither the `Separations` or `Fabrication` facilities have throughput constraints, i.e., both facilities are immediately able to process any quantity of fuel. Such constraining values can (and should) be added by users in order to obtain more realistic behavior; however, they are excluded in this paper in order to simplify the underlying dynamics of the processes being investigated.

A comparison of the inventories of plutonium (Pu) in each facility type of interest among the `base-case` and `outage` scenarios is shown in Figure 7. As can be seen in Figure 7a, the quantity of MOX in `Reactor`s is under a dynamic equilibrium, oscillating between the maximum quantity allowable in the system and one refueling quantity less than the maximum, based on refueling schedules. The equilibrium value increases in a stair-step-function manner as the number of reactors increases to being able to provide sufficient used UOX for the next marginal refueling quantity of MOX. The quantity of MOX in `Fabrication` oscillates between a minimal value and a maximum value which is sufficient for a single reactor's refueling quantity. As soon as there is sufficient MOX fuel for another refueling *and* a reactor makes a request to be refueled, it is provided the quantity of MOX in of fresh fuel. Finally, `Separations` separates the various actinides of used fuel and passes on fissile isotopes to `Fabrication`, thus maintaining a small oscillating inventory in each timestep.

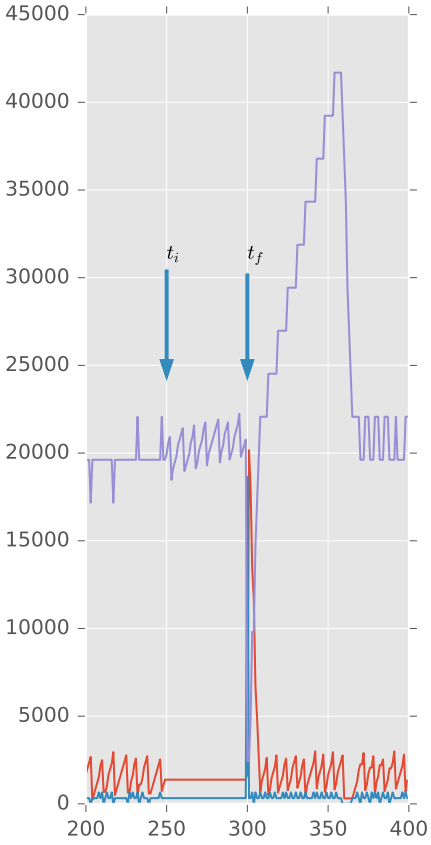
The dynamic equilibrium behavior changes in the `outage` scenario after the initial outage time, $t_i$, as is observable in Figures 7b and 7c. Because the outage occurs in `Separations`, which takes input from the `Reactor`s and provides output to `Fabrication`, the inventories of both `Separations` and `Fabrication` remain constant for the duration of the outage period. The inventory of Pu in `Reactor`s continues to oscillate because MOX assemblies are discharged and continue to be sent to `Storage`, whereas spent UOX assemblies (with significant Pu inventories) are stored on site. In the first timestep of renewed service of `Separations`, $t_f$,




(a) Pu inventories in the `base-case` scenario.




(b) Pu inventories in the `outage` scenario.




(c) A close-up of the `outage` scenario perturbation.

Figure 7: Facility inventories of Pu in `base-case` and `outage` scenarios.

the entirety of the pent-up store of used fuel in `Reactor`s is send to `Separations`, reducing the inventory to zero, causing the delta-function behavior in `Reactor` flows seen in Figure 7c at $t = t_f$. `Separations` then extracts all of Pu in a single timestep, sending it to `Fabrication` and causing the delta-function behavior in `Separations` flows seen in Figure 7c at $t = t_f + 1$. Finally, the stock of Pu in `Reactor`s after the outage increases due to the higher availability of MOX fuel in `Fabrication`, until the dynamic equilibrium returns. The length of the perturbation is function of both the amount of Pu required per refueling and the number of refuelings that occurs during the outage. The more refuelings that happen during the outage, the more excess MOX assemblies can be made, thus continuing the dynamic equilibrium perturbation.

### *3.2. External MOX Supplier: Fuel Cycles with Demand Fungibility*

The DRE allows for both positive and negative perturbations in fuel availability. While the `outage` scenario models a case where there is a supply-chain disruption, the `external` scenario models a case where there is an injection of a preferred commodity source. An example of such a scenario occurring in the real world includes the down-blending of military-grade fuel sources, such as the MOX Fuel Fabrication Facility, in which a preferred fuel commodity is introduced, and the Megatons to Megawatts (MT2MW) program, where a preferred fuel fabrication commodity is introduced at the enrichment-fabrication facility interface.

In the `external` scenario, an external source of MOX fuel enters halfway through the simulation at $t = 250$, creating the fuel cycle shown in Figure 8. The total quantity of fuel the external source can provide is limited to 10 refueling quantities (where reactors refuel one third of their total core mass in each cycle). Preferences are assigned such that reactors prefer MOX from its normal cycle over MOX from the external source, i.e., $p_{\text{MOX}} > p_{\text{MOX, external}} > p_{\text{UOX}}$. Reactors request each of the commodities, and thus the first 10 reactors to refuel after the external source enters the simulation when no original MOX is available will be provided with MOX from the external facility. Reactors will continue to request

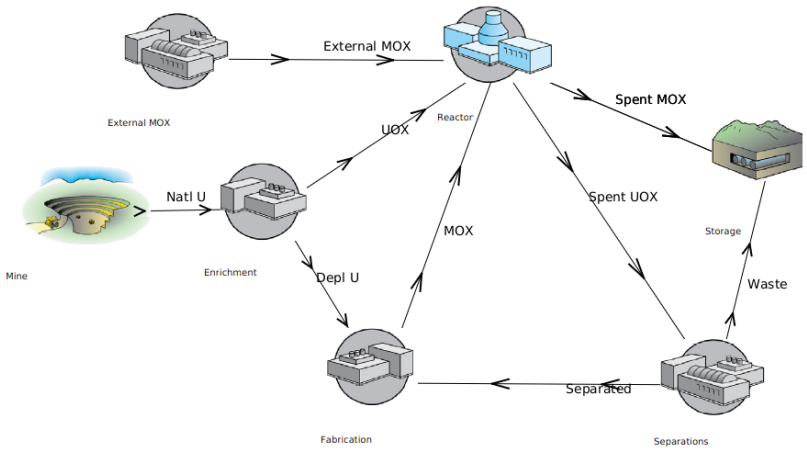



Figure 8: Material routing between in the `external` scenario fuel cycle. Possible arc flows are labeled with commodity names.

fuel from the external facility for the remainder of the simulation, but will not receive any due to the limited total inventory. This injection of a new fuel source also serves to perturb the supply chain by delaying the amount of spent UOX available for recycling.

The dynamic equilibrium of Pu inventories again changes with the `external` perturbation, as shown in Figure 9. A number of new features arise, however. First, the equilibrium value during the initial transient increases by the total quantity of refueling quantities available from the external source of MOX (in this case 10 refueling quantities). Second, the equilibrium value upon exiting the transient is lower than the value upon its entrance. This is due to the fact that the amount of spent UOX in the overall recycle system has decreased, due to the usage of external MOX, thus reducing the availability of MOX. The system has been shocked into a new dynamic equilibrium, with Pu values slightly lower than the previous equilibrium. This suggests that the injection of external recycled fuel can reduce the level of which a system can sustain a recycling fuel cycle.

Finally, a small lag can be seen in the inventory of Pu in `Fabrication`, which is due to a loss of available spent UOX due to the increased presence of spent MOX exiting reactors that were able to utilize external MOX. The Pu inventories recover quickly from this transient, however.

### *3.3. Regional Tariffs: Fuel Cycles with International Instruments*

One of the novel features of the DRE is the ability for different geographical and managing entity representations to be laid over otherwise regional-agnostic fuel cycles and affect the outcome of possible trades between those fuel cycles. The `tariff` two-region scenario showcases the ability to model such situations.

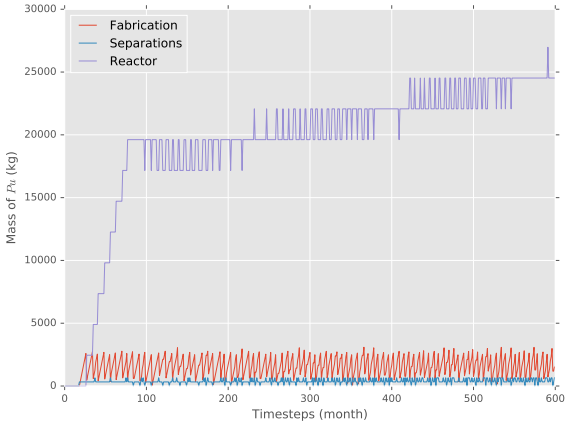
Two regions, Region A and Region B, are modeled. Region A houses a fuel cycle with both UOX and MOX-based fuel services, as in the `base-case` scenario. The same total number of `Reactor`s are modeled in the scenario. Region A begins with 15 `Reactor`s and Region B begins with 5 reactors. All reactor deployment occurs in Region A as shown in Figure 5.

In this scenario, Region A can provide UOX and MOX fuel services to other regions using a fuel take-back model (all fuel provided as a service is returned after it has been used in a reactor). Repatriation of fission products to the lessee region is not modeled in this scenario for purposes of clarity. Region B contains a simple, once-through fuel cycle. Although the scenario is somewhat contrived in order to highlight a multi-commodity system under dynamic behavioral change, such fuel service arrangements are present today in countries that provide fuel for once-through fuel cycles, e.g., Russia [35]. The possible flow of commodities between fuel cycles is shown in Figure 10.

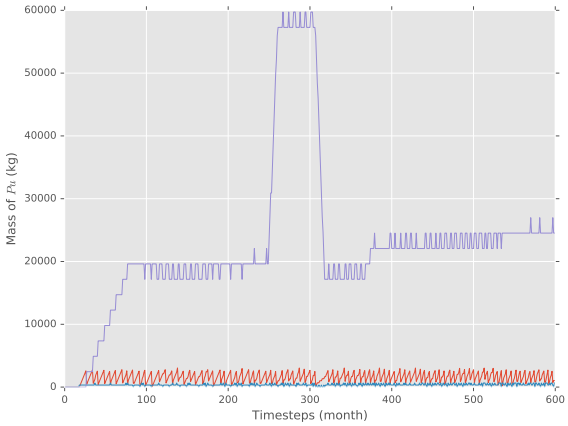
Initially, preferences are set such that fuel trade from Region A to Region B is preferred over Region B's domestic fuel production. In other words, a preference distribution for fuel supplied to Region B has the following relation

$$p_{MOX,a} > p_{UOX,a} > p_{UOX,b} > 1. \tag{27}$$

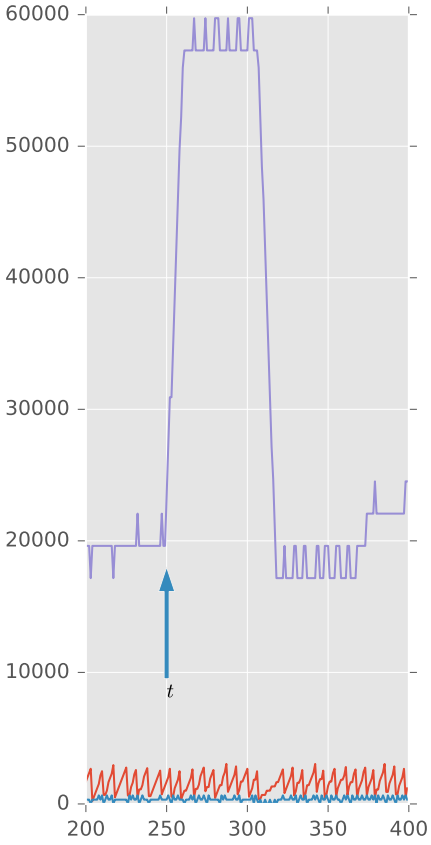
This preference distribution implies that Region B's domestic fuel cycle will never be utilized – it will always be fueled by Region A, as long as Region A has




(a) Pu inventories in the `base-case` scenario.




(b) Pu inventories in the `external` scenario.




(c) A close-up of the `external` scenario perturbation.

Figure 9: Facility inventories of Pu in `base-case` and `external` scenarios.

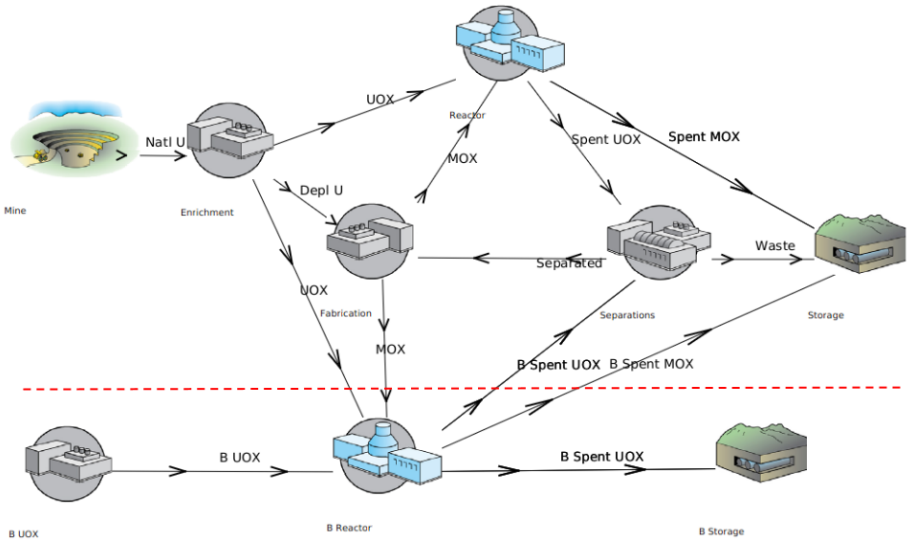



Figure 10: A two-region set of fuel cycles separated by a dotted-red line. The upper region (Region A) includes a one-pass MOX fuel cycle, and the bottom region (Region B) includes a once-through fuel cycle connected to the one-pass MOX fuel cycle. Note that all spent fuel that originated in Region A is returned to Region A's fuel cycle.

available capacity.

At some time $t_0$, a time-varying tariff is applied by Region B which perturbs preference values along arcs connecting Region A fuel suppliers with Region B fuel consumers. Consider a tariff defined by function in Equation 28 with preferences adhering to the relation provided in Equation 29, which guarantees a strict preference ordering under $f(t)$.

$$f(t) \begin{cases} 1, & \text{if } t < t_0 \\ \frac{p_{UOX,b}-1}{p_{UOX,a}}, & \text{if } t_0 \leq t < t_1 \\ \frac{p_{UOX,b}-1}{p_{MOX,a}}, & \text{if } t_1 \leq t < t_2 \end{cases} \tag{28}$$

$$p_{UOX,b}\left(1 - \frac{p_{UOX,a}}{p_{MOX,a}}\right) > 1. \tag{29}$$

Choosing nominal values that satisfy Equations 27 and 29, e.g., $p_{MOX,a} = 9$, $p_{UOX,a} = 4$, and $p_{UOX,b} = 2$, one arrives at actual preference values as shown in Figure 11. In the `tariff` scenario, $t_0$ is chosen to be 150 and $t_1$ is set to 300. The DRE naturally handles the flow of commodities between `Facility` agents in each `Region`, allowing the application of tariffs by the `Region` agents.

The effects of time-dependent tariff application on the simulation can be seen in Figure 12. The front-end of Region B's fuel cycle is not utilized until $t > t_0$; all fuel is provided from Region A. After $t_0$, the majority of the fuel services required by `Reactor`s in Region B comes from Region B's own front end. However, it is still able to utilize the MOX-based fuel services from Region A. Finally, after the final tariff is applied at $t_1$, Region B's `Reactor`s stop utilizing Region A's fuel cycle entirely.

### *3.4. Comparisons between Scenarios*

In each scenario, the total amount of electricity generated is identical in order to compare the mechanics and results of fuel supply and demand. Therefore, comparisons between all scenarios is most easily made by observing the total fuel usage by reactors for each commodity type. A summary of this metric is provided in Figure 13. Figure 13a displays the cumulative flow of UOX fuel into

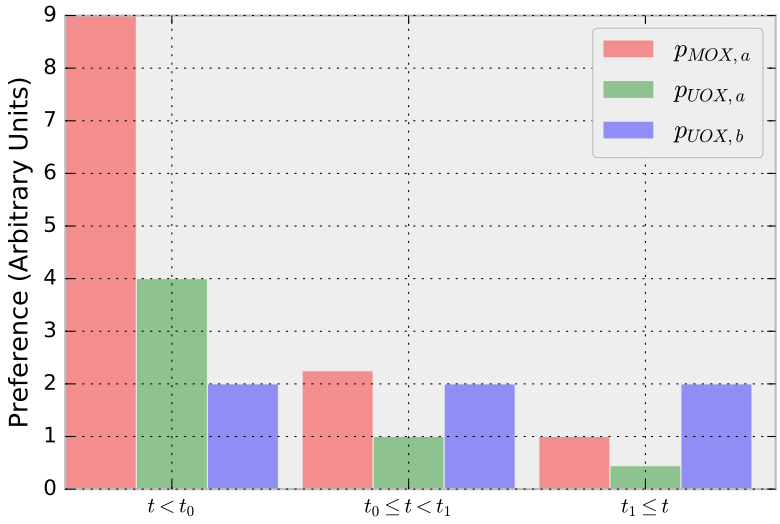



Figure 11: Preference values for reactors in Region B for available fuel commodities in Region B as a function of time.

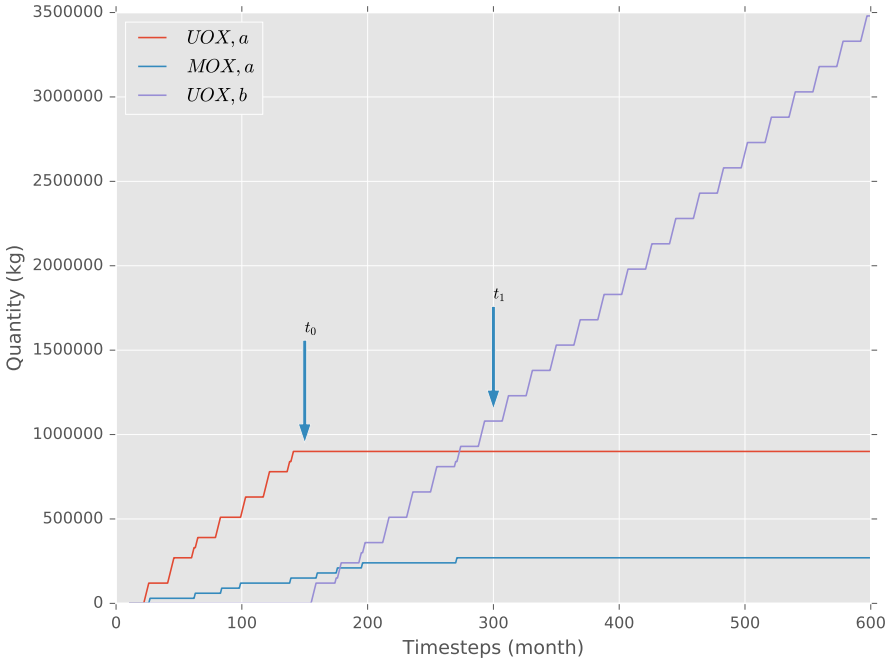
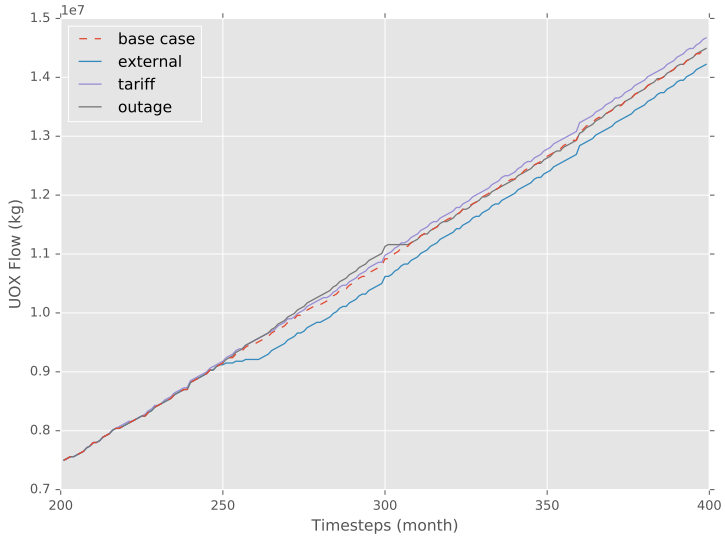


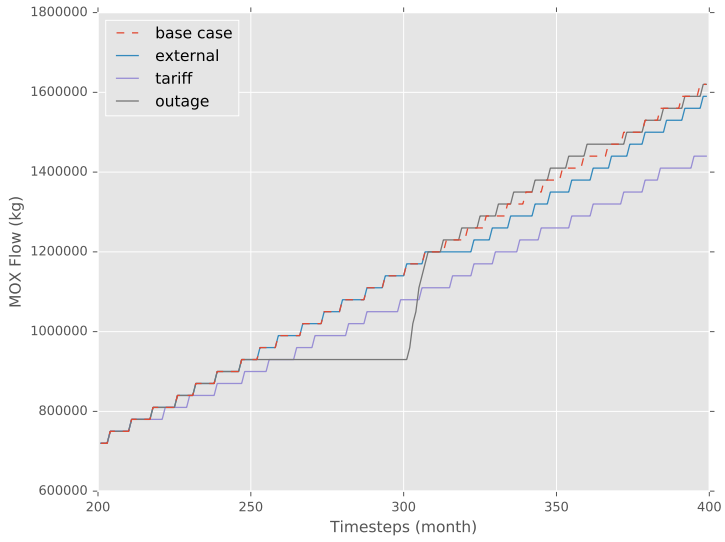

Figure 12: Cumulative flow of fuel into `Reactors` in Region B.

reactors over all scenarios which shows a number of interesting effects. First, the `external` scenario has the lowest cumulative flow, which is expected of a scenario in which an external source of non-UOX is provided. The `outage` scenario initially deviates from the `base-case` scenario, but eventually returns to match its cumulative UOX usage. This is due to the fact that in the defined system, outages simply serve to store fuel. When sufficient time has past after the outage, the system returns to its dynamic equilibrium. Finally, the `tariff` scenario utilizes the most UOX fuel of all, providing an interesting case study of the effects of dynamic equilibrium. In the `tariff` scenario, there is a population of reactors that prefers UOX fuel that is not sent to `Separations` (i.e., the UOX fuel in its own Region) over all other sources in the simulation. Therefore, the population of Pu is reduced, which reduces both the quantity and frequency of MOX availability. This, in turn, increases overall UOX consumption. In short, MOX availability decreases due to upstream supply chain effects, causing an increase in overall UOX consumption.

Figure 13b showcases the cumulative flow of MOX fuel into all reactors as a function of simulation time step. Because reactors can be fueled only with UOX or MOX, it represents the inverse of Figure 13a. For example, whereas the `tariff` scenario utilizes the most UOX, it utilizes the least MOX for the same reasons. A number of additional features can be observed in Figure 13b dealing with departures from dynamic equilibrium of the `base-case` scenario. The undershooting and then overshooting of MOX consumption in the `outage` scenario is visible. During the outage, less MOX is consumed, but immediately after the outage, excess MOX is consumed until there is a return to dynamic equilibrium. Additionally, a reduction in the total amount of MOX (sourced from recycled UOX) consumed is observed in the `external` scenario. This is due to a reduction in the available recycled UOX supply during periods of external MOX consumption. In short, for each reload of external MOX, the system loses a future amount of recyclable UOX.




(a) UOX flow.




(b) MOX flow

Figure 13: Cumulative flow of fuel in all Scenarios. The timestep period between 200 and 400 is chosen to highlight all relevant transients.

Table 6: Solution times required for full simulation runs of each solver type in each scenario. Simulations were performed on a Macbook Air with a duel-core i5 2.6 GHz processor using a Ubuntu 14.04 operating system.

| Scenario | Solver | Time (s) |
|---|---|---|
| `base-case` | cbc | 6.306 |
| | greedy | 4.169 |
| `external` | cbc | 6.371 |
| | greedy | 3.149 |
| `outage` | cbc | 5.978 |
| | greedy | 3.161 |
| `tariff` | cbc | 6.074 |
| | greedy | 3.079 |

### *3.5. Solver Comparisons*

Each of the above scenarios was executed by solving the DRE using both the `Greedy` heuristic and full optimization (results are shown from the `Greedy` heuristic cases for clarity). Differences in simulation time are observed, as shown in Table 6. Solving the DRE with full optimization takes approximately twice as long as solving it with the `Greedy` heuristic in these simulations. These are small simulations and one can expect this solution time gap to increase exponentially with simulation size, as solutions to MILPs increase exponentially with time and the `Greedy` heuristic is a polynomial-time algorithm. The primary component of simulation size that is of concern to solution time is the total number of supply and request nodes in each problem. These will increase with the number of entities in the simulation, the number of commodities, and the connectedness of entities (i.e., the number of potential transfers between entities).

Each of the results in Table 6 is with respect to the simple scenarios presented in this paper. In order to ascertain a sense of how each solver will work with larger scenarios, the `base-case` scenario was run an additional time after modifying the reactor deployment schedule as follows: in each instance when a single

reactor would be deployed, five reactors are deployed instead, increasing the total reactor population by a factor for five to approximately 150. The greedy and cbc `base-case` scenarios solved in 10.7 and 20.8 seconds, respectively. This implies that there is not a clear linear relationship between solution time of a full simulation run and the number of entities in the simulation for either solver – there are likely additional simulation dynamics occurring outside of the DRE that affect solution times. Additionally, this simple exercise shows that simulations of reactor populations approximately the size of the United States reactor fleet can be solved in very reasonable times.

Interestingly, slight differences in simulation results was also observed between the two solvers. This is due to time-steps in which there is problem degeneracy, i.e., where multiple optimal solutions exist. Consider a timestep, $t_i$, in which one reactor is refueling and another reactor is entering the simulation. Four total quantities of fuel are requested - one refueling batch and three initial core batches. Consider further that one batch of MOX fuel is available. Because reactors all make requests to the DRE with identical preferences, there are four degenerate optimal solutions – one for each potential assignment of the MOX batch. This leads to three potential simulation futures, one in which the MOX batch is ejected after one cycle (if it assigned to the refueling reactor or to the initial spot of the new reactor), one in which the MOX batch is ejected after two cycles, and a third after three cycles. In each simulation future, the quantity of fuel in the recycle loop at $t > t_i$ is different, causing differing future behavior and overall simulation results.

## 4. Conclusions

A hybrid simulation-optimization approach to the dynamic modeling of the NFC has been presented and implemented in the Cyclus NFC simulator. Focus has been placed on separating the core simulator design and the associated entities forming the simulation. Significant constraints were placed on the design of an entity interaction mechanism. It must support arbitrary physics and chemical

constraints, as well as general supply-chain constraints, such as inventory and processing constraints. Further, it must model the competition of resources among entities for which demand and supply of resources may be fungible. Finally, inter-entity interactions must be translated to the interaction framework. The resulting interaction mechanism, termed the Dynamic Resource Exchange (DRE), was informed chiefly from the fields of supply-chain management, agent-based modeling, and mathematical programming.

The DRE allows agents to inform both system supply and demand of resources through a request-bid framework. Physics fidelity is provided to agents in this framework by utilizing fully specified `Resource` objects. For example, nuclear fuel demand can be specified directly by an ideal isotopic vector in a `Material` object. Once supply and demand is known, social interaction models can be applied to affect resource flow-driving mechanisms. For example, a tariff can be modeled by uniformly reducing preferences of transactions between agents outside of a given `Region`. Presently, a cardinal preference model is used as the flow-driving mechanism.

The DRE comprises three layers: a resource layer, with which agents interact, an exchange layer, and a formulation layer. Supply, demand, and preferences are defined in the resource layer, for a specific type of `Resource` object. The exchange layer provides a general resource exchange representation, irrespective of a specific object type. The representation comprises a bipartite graph of supply and demand nodes, supply and demand constraints, and a measure of preference for each proposed connection between nodes. The *exchange graph* generated by the DRE is solved directly by a heuristic or translated into the NFCTP, a multicommodity transportation problem, and solved accordingly. Resulting trades between entities are then communicated back to the simulation.

The ability to include socioeconomic, agent-based interactions in NFC simulation were demonstrated using the `Tariff Region`, a new entrant in the Cyclus ecosystem. Further, scenarios that highlight unique modeling aspects enabled by used of the DRE were shown. The `external` scenario highlighted system response to dynamic entry and exit of an external source of recycled fuel. The `outage`

case displayed how a fuel cycle system responds to unplanned individual facility outages. Finally the `tariff` scenario demonstrated how sociopolitical models can be overlaid on top of existing fuel cycle models materially affecting scenario results. Finally, the effects of using full optimization rather than heuristics was investigated.

Each of the scenarios described in section 3 intentionally highlights one unique aspect of novel modeling capability. Critically, each new modeling feature is modular and can be combined with any other. The core flexibility that the DRE provides is a common communication and solution framework. Analysts instead can focus on the specifics of their NFC core competency. A common set of rules, specialized for each simulation entity, define its interaction mechanism, rather than deciding *a priori* exactly how entities interact. This flexibility allows for dynamic entity exit and entry into the simulation as well as the ability to model stochastic events.

Using a simulator that employ the DRE, e.g., Cyclus, users may now apply physical, economic, and social models to NFC simulation. The choice of solver will largely depend on the fidelity of the associated models and underlying data. The `Greedy` solver will always provide a feasible solution to the given exchange instance, applying any physical, chemical, or supply-chain constraints. Therefore, if a user has a low-fidelity economic or social model, then the Greedy solver will likely meet the users needs. With higher-fidelity economic and social models, obtaining a optimal solution becomes paramount. `COIN-CBC` is an available resource for solving such instances, though commercial solvers such as CPLEX are much more computationally efficient. Solvers, in principle, can be easily exchanged because of the use of OSI.

A novel way to model dynamic, nuclear fuel cycles has been proposed, designed, implemented, and presented. New features include competition between suppliers and consumers, constrained supply and consumption, and the inclusion of extra-facility effects, such as state-level relationships. Ongoing work continues to utilize the flexible, entity-interaction mechanism enabled by the DRE in order to support modeling of nonproliferation applications and medium-fidelity physics-

enabled fuel cycles. The implementation of the DRE in Cyclus represents a significant methodological advance in NFC simulation, supporting a variety of existing and new simulation techniques, each within a common simulator framework.

**Acknowledgements**

The authors would like to thank the Nuclear Engineering University Program (NEUP) for their generous support in the form of both an Integrated University Program (IUP) fellowship as well as the grants DOE-NE00120341 and DOE-NE0000673. The authors would also like to thank all developers of the Cyclus simulator, especially Kyle Oliver, Dr. Kathryn Huff, Dr. Anthony Scopatz, Dr. Robert Carlsen, and Arrielle Opotowsky. Finally, the authors would also like to thank Drs. James Luedtke, Michael Corradini, Laura McLay, and Erich Schneider for their comments on the development of the methodology of the paper.

## References


[1] N. R. Brown, B. W. Carlsen, B. W. Dixon, B. Feng, H. R. Greenberg, R. D. Hays, S. Passerini, M. Todosow, A. Worrall, Identification of fuel cycle simulator functionalities for analysis of transition to a new fuel cycle, Annals of Nuclear Energy 96 (2016) 88–95.

[2] R. Busquim e Silva, M. Kazimi, P. Hejzlar, A system dynamics study of the nuclear fuel cycle with recycling: Options and outcomes for the US and brazil, Tech. Rep. MIT-NFC-TR-103, MIT Center for Advanced Nuclear Energy Systems (CANES), Cambridge, MA, United States (Nov. 2008).

[3] L. V. D. Durpel, A. Yacout, D. Wade, H. Khalil, Daness dynamic analysis of nuclear system strategies, in: Global 2003: Atoms for Prosperity: Updating Eisenhouwer's Global Vision for Nuclear Energy, American Nuclear Society, New Orleans, LA, United States, 2003, pp. 1613–1620.

[4] A. Yacout, J. Jacobson, G. Matthern, S. Piet, D. Shropshire, C. Laws, Vision–verifiable fuel cycle simulation of nuclear fuel cycle dynamics, Waste Management.

[5] A. Worrall, R. Gregg, Scenario Analyses of Future UK Fuel Cycle Options, Journal of Nuclear Science and Technology 44 (3) (2007) 249–256. doi:10.1080/18811248.2007.9711279.
URL http://www.tandfonline.com/doi/abs/10.1080/18811248.2007.9711279

[6] E. A. Schneider, C. G. Bathke, M. R. James, NFCSim: A dynamic fuel burnup and fuel cycle simulation tool, Nuclear Technology 151 (1) (2005) 35–50.

[7] L. Boucher, J. P. Grouiller, "COSI": the complete renewal of the simulation software for the fuel cycle analysis, in: Fuel Cycle and High Level Waste Management, Vol. 1, ASME, New York, NY, USA, Miami, FL, United States, 2006.

[8] B. Mouginot, J. Clavel, N. Thiolliere, Class, a new tool for nuclear scenarios: Description & first application, World Academy of Science, Engineering and Technology, International Journal of Mathematical, Computational, Physical, Electrical and Computer Engineering 6 (3) (2012) 232–235.

[9] T. M. Schweitzer, Improved building methodology and analysis of delay scenarios of advanced nuclear fuel cycles with the verifiable fuel cycle simulation model (VISION).
URL http://repository.lib.ncsu.edu/ir/handle/1840.16/123

[10] V. D. Durpel, A. Yacout, D. Wade, T. Taiwo, U. Lauferts, DANESS v4.2: Overview of capabilities and developments, in: Proceedings of Global 2009, Paris, France, 2009.

[11] L. Guerin, M. Kazimi, Impact of alternative nuclear fuel cycle options on infrastructure and fuel requirements, actinide and waste inventories, and economics, Technical Report MIT-NFC-TR-111, MIT Center for Advanced Nuclear Energy Systems (CANES), Cambridge, MA, United States (Sep. 2009).

[12] J. W. Forrester, Counterintuitive behavior of social systems, Theory and Decision 2 (2) (1971) 109–140.

[13] A. M. Law, D. M. Kelton, Simulation Modeling and Analysis, 3rd Edition, McGraw-Hill Higher Education, 1999.

[14] K. D. Huff, M. J. Gidden, R. W. Carlsen, R. R. Flanagan, M. B. McGarry, A. C. Opotowsky, E. A. Schneider, A. M. Scopatz, P. P. Wilson, Fundamental concepts in the cyclus nuclear fuel cycle simulation framework, Advances in Engineering Software 94 (2016) 46 – 59. doi:http://dx.doi.org/10.1016/j.advengsoft.2016.01.014.
URL http://www.sciencedirect.com/science/article/pii/S0965997816300229

[15] R. W. Carlsen, P. P. Wilson, Challenging fuel cycle modeling assumptions: Facility and time step discretization effects, Nuclear Technology 195 (3).
URL dx.doi.org/10.13182/NT15-138

[16] J. Swaminathan, S. Smith, N. Sadeh, Modeling supply chain dynamics: A multiagent approach, Decision Sciences 29 (3) (1998) 607–632.

[17] N. Julka, R. Srinivasan, I. Karimi, Agent-based supply chain management-1: framework, Computers & Chemical Engineering 26 (12) (2002) 1755–1769.

[18] D. J. Van der Zee, J. Van der Vorst, A modeling framework for supply chain simulation: Opportunities for improved decision making, Decision Sciences 36 (1) (2005) 65–95.

[19] D. C. Chatfield, J. C. Hayya, T. P. Harrison, A multi-formalism architecture for agent-based, order-centric supply chain simulation, Simulation Modelling Practice and Theory 15 (2) (2007) 153–174.

[20] J. Holmgren, P. Davidsson, J. A. Persson, L. Ramstedt, An agent based simulator for production and transportation of products, in: The 11th World Conference on Transport Research, Berkeley, USA, 2007, pp. 8–12.

[21] M. J. Gidden, An Agent-Based Modeling Framework and Application for the Generic Nuclear Fuel Cycle, Prelim, University of Wisconsin, Madison (Sep. 2013).
URL http://dx.doi.org/10.6084/m9.figshare.1132596

[22] M. Gidden, R. Carlsen, A. Opotowsky, O. Rakhimov, A. Scopatz, P. Wilson, Agent-based dynamic resource exchange in Cyclus, in: Proceedings of PHYSOR, Kyoto, Japan, 2014.

[23] M. Gidden, P. Wilson, An Agent-Based Framework for Fuel Cycle Simulation with Recycling, in: Proceedings of GLOBAL, Salt Lake City, UT, United States, 2013.

[24] S. Even, A. Itai, A. Shamir, On the complexity of time table and multi-commodity flow problems, in: Foundations of Computer Science, 1975., 16th Annual Symposium on, IEEE, 1975, pp. 184–193.

[25] J. Forrest, et al., COIN-OR Open Solver Interface, `https://projects.coin-or.org/Osi` (Dec. 2014).
URL `https://projects.coin-or.org/Osi`

[26] R. H. Strotz, Cardinal utility, The American Economic Review (1953) 384–397.

[27] K. M. Oliver, Geniusv2: Software design and mathematical formulations for multi-region discrete nuclear fuel cycle simulation and analysis, Ph.D. thesis, University of Wisconsin-Madison (2009).

[28] D. E. Shropshire, K. A. Williams, W. B. Boore, J. D. Smith, B. W. Dixon, M. Dunzik-Gougar, R. D. Adams, D. Gombert, Advanced fuel cycle cost basis, Tech. rep. (Aug. 2009).

[29] R. Mazumdar, L. Mason, C. Douligeris, Fairness in network optimal flow control: optimality of product forms, Communications, IEEE Transactions on 39 (5) (1991) 775–782. `doi:10.1109/26.87140`.

[30] A. Nagurney, J. Dong, D. Zhang, A supply chain network equilibrium model, Transportation Research Part E: Logistics and Transportation Review 38 (5) (2002) 281 – 303. `doi:10.1016/S1366-5545(01)00020-5`.

[31] H. Song, C.-C. Liu, J. Lawarree, Nash equilibrium bidding strategies in a bilateral electricity market, Power Systems, IEEE Transactions on 17 (1) (2002) 73–79. `doi:10.1109/59.982195`.

[32] J. Forrest, Cbc (coin-or branch and cut) open-source mixed integer programming solver, 2012, URL https://projects. coin-or. org/Cbc.

[33] E. A. Andrianova, V. D. Davidenko, V. F. Tsibul'skii, Desae program for systems studies of long-term growth of nuclear power, Atomic energy 105 (6) (2008) 385–390.

[34] M. Gidden, P. Wilson, GitHub repository: dre-paper, https://github.com/cyclus/dre-paper (2016).

[35] World Nuclear Association: Russia's Nuclear Fuel Cycle, http://www.world-nuclear.org/information-library/country-profiles/countries-o-s/russia-nuclear-fuel-cycle.aspx, accessed: 2016-06-05.